

129

E 10

TK 55.234

K

KFKI-1977-27

Д. Амбрози
Ф. Вайда
И. Рени

ПРИНЦИПЫ ПОСТРОЕНИЯ И ОБЛАСТИ ПРИМЕНЕ-
НИЯ ИНТЕЛЛЕКТУАЛЬНОГО ДИСПЛЕЙНОГО ТЕР-
МИНАЛА С МИКРОПРОЦЕССОРНЫМ УПРАВЛЕНИЕМ

Hungarian Academy of Sciences

CENTRAL
RESEARCH
INSTITUTE FOR
PHYSICS



1977 JUN 17

BUDAPEST

2017

ПРИНЦИПЫ ПОСТРОЕНИЯ И ОБЛАСТИ ПРИМЕНЕНИЯ ИНТЕЛЛЕКТУАЛЬНОГО
ДИСПЛЕЙНОГО ТЕРМИНАЛА С МИКРОПРОЦЕССОРНЫМ УПРАВЛЕНИЕМ

Д.Амбрози, Ф.Вайда и И.Рени

Центральный институт физических исследований ВАН Будапешт

Отправлено на Международный симпозиум
ядерной электроники, Варна, Болгария,
5-9 мая 1977 г.

АННОТАЦИЯ

Описывался интеллектуальный дисплейный терминал с микропроцессорным управлением, работающий по принципу стандартной телевизионной растровой системы. Так как терминал первоначально был разработан для управления процессами в промышленности в ходе проектирования придавалось большое внимание модульному построению и гибкости адаптации. Разработка строительных элементов аппаратной, аппаратно-программной (firmware) и программной систем осуществлялась на основе оптимального разделения задачи. Коротко излагалось построение, работа отдельных строительных элементов и в конце приводятся несколько примеров для возможностей применения терминала.

ABSTRACT

The paper presents a microprocessor controlled intelligent display terminal based on the standard TV raster principle. As the terminal was designed mainly for industrial process control applications, particular attention was devoted to modularity, flexibility and adaptability. The hardware /firmware/ software tradeoffs were determined according to the optimal division of tasks. After the discussion of the hardware, software and firmware modules of the terminal a few examples of the terminal's application are shown.

KIVONAT

A cikkben egy intelligens megjelenítő terminált ismertetünk, amely mikroprocesszoros vezérléssel, szabványos TV-raszter elven működik. Mivel a terminál elsődlegesen ipari folyamatirányításokhoz készült, a tervezés során nagy súlyt helyeztünk a modularitásra, rugalmasságra, adaptálhatóságra. A hardware-, firmware- és software építőelemek kialakítása az optimális feladatfelosztás szempontjai alapján történt. Röviden bemutatjuk az egyes alkotóelemek felépítését, működését, végül néhány példával világítunk rá a terminál alkalmazási lehetőségeire.

ПРИНЦИПЫ ПОСТРОЕНИЯ И ОБЛАСТИ ПРИМЕНЕНИЯ ИНТЕЛЛЕКТУАЛЬНОГО
ДИСПЛЕЙНОГО ТЕРМИНАЛА С МИКРОПРОЦЕССОРНЫМ УПРАВЛЕНИЕМ

Д.Амбрози, Ф.Вайда и И.Рени

Центральный Институт физических исследований ВАН Будапешт

1. Введение

В ЦИФИ уже давно занимаемся вопросами применения ЭВМ. В начале сбор данных в реальном масштабе времени явился самым важным направлением исследования, а потом, когда и связь оператора с ЭВМ стала важной областью исследования, мы начали интенсивно заниматься решением задач, касающихся связи оператора с ЭВМ. В настоящее время вопросы сбора данных в лабораториях перерастает в вопросы управления процессами в промышленности, а в этой области связь оператора с процессами имеет большое значение. Поэтому уже давно занимаемся разработкой методов и оборудования связи оператора с ЭВМ.

По стандартным телевизионным методам и принципу раstra были разработаны разные растровые телевизионные системы, которые были успешно применены в разных областях. Когда поднялся вопрос о создании современной связи оператора с процессами, дальнейшая разработка дисплеев оказалась очевидной. Конечно, такие методы применения имеют и специальные требования. Терминал должен отображать функциональную схему процесса или только одну часть процесса, по которым видны мгновенные значения данных, поступивших от различных преобразователей измерения, и видно текущее состояние средств вмешательства (например в случаешибера - закрытое, полужакрытое или полностью открытое состояние). Так, терминал дает оператору удовлетворительную информацию для вмешательства в

данном положении; в специальном случае же (например в случае погрешности) оказывает помощь поиску причины ошибки изображением данных с достаточной подробностью и в соответствующей форме.

Чтобы немного подробнее рассматривать построение такой системы, можно дать общую схему системы.

На рис. 1 приведена простая система управления технологическими процессами с ЭВМ. Изображение на дисплейном терминале (например схема процесса) может быть разделено на две части:

- 1.) информацию, постоянную в ходе процесса; например: блок-схема управляемого процесса;
- 2.) данные и состояния средств вмешательства, представленные со символами, которые в ходе процесса изменяются.

На рисунке видно, что постоянная информация изображения может быть запомнена на магнитном диске, изменяющаяся информация изображения получается программой, контролирующей и управляющей процессором и выполняемой ЭВМ. О процессе или о его отдельных частях во внешней памяти могут быть запомнены порядка нескольких десятков рисунков и они могут быть воспроизведены в любое время.

Параллельно программе реального времени в ЭВМ работает и программа обслуживания терминала (handler), которая обеспечивает вызов информации изображения и после произведения измерений направляет изменяющиеся значения для терминала. В случае надобности, имеется возможность для вмешательства оператора через терминал. В этом случае команда оператора (с консоли) передается программой обслуживания терминала для программы реального времени. (Рис. I).

На рис. 2 приведена одновременная работа двух ЭВМ. ЭВМ, управляющая процессами сообщает об изменяющейся информации изображения малой конеч-

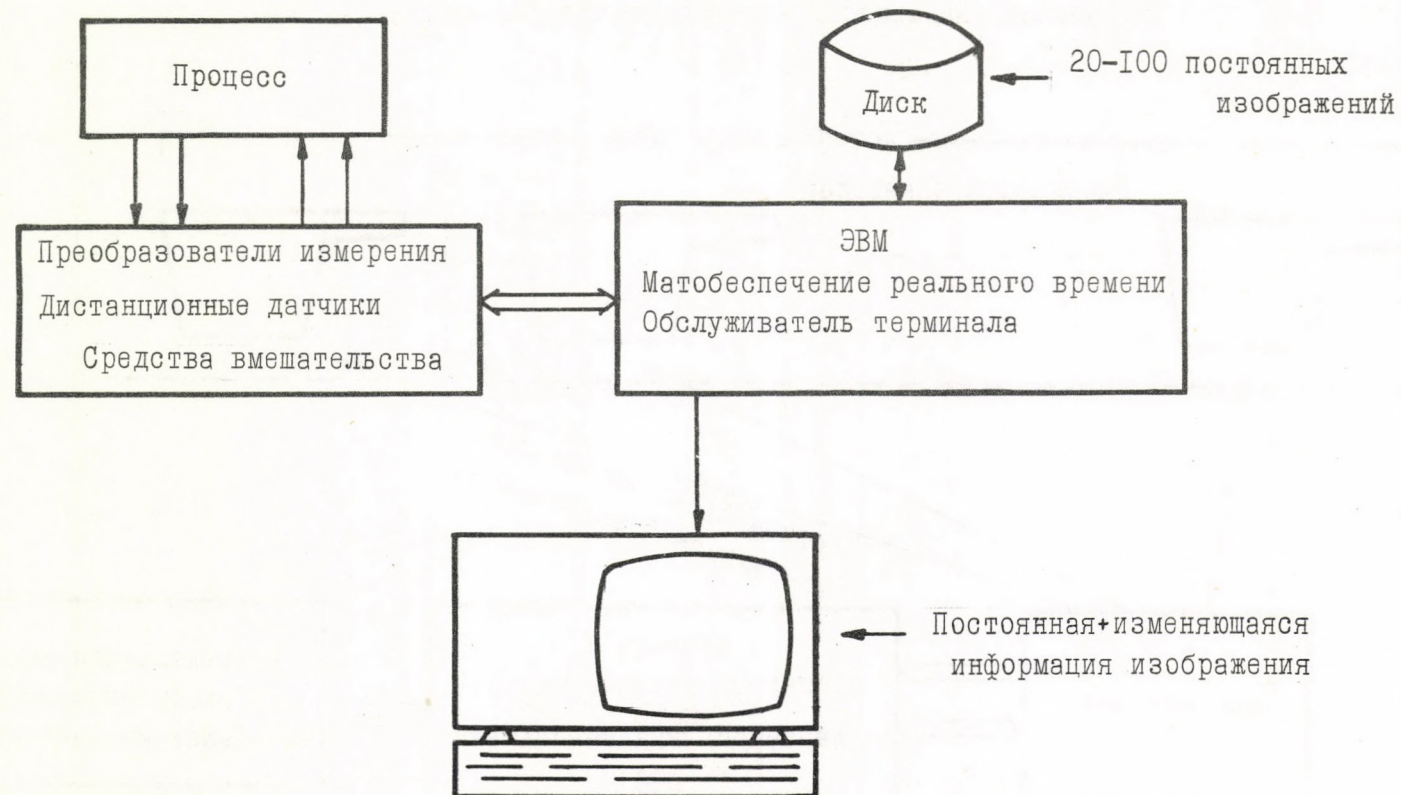


Рис. I Система управления технологическими процессами с дисплейным терминалом

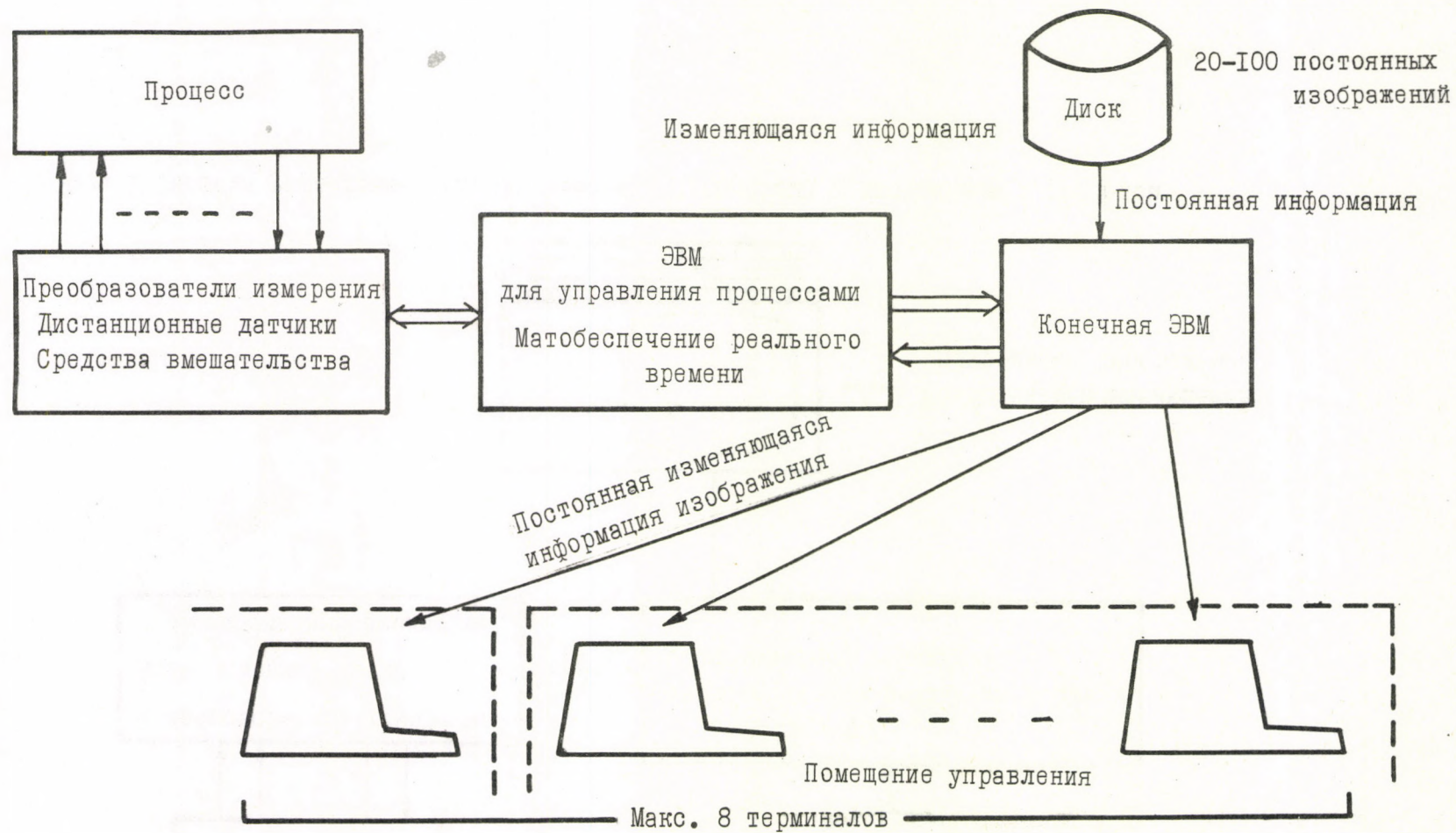


Рис. 2 Многотерминальная система управления процессами

ной ЭВМ, обслуживающей многие терминалы, а постоянную информацию малая конечная ЭВМ получает от диска. В одно время на экранах макс. 8 дисплейных терминалов могут быть показаны разные этапы процесса, данные дневника или другая информация; по желанию может быть вызвано любое из изображений.

Учитывая тот факт, что терминал должен быть пригодным не только для контроля и управления одним специальным промышленным процессом, но и какими угодно процессами, при разработке терминала наша основная идея была: гибкость и возможность адаптации.

Некоторые характеристики, зависящие от процесса приводим здесь:

- тип отображаемой информации (например, алфавитноцифровая, функциональная схема, диаграмма, гистограмма, их комбинация);
- сложность управляемого процесса (начиная от управления измерениями в лаборатории до управления комплексными системами);
- тип ЭВМ, выполняющий управление процессами;
- программная система для управления процессами.

Создание терминала, независимого от применения, было определено следующими требованиями:

- применение символов различного типа и разной сложности (возможность нового определения и увеличения количества символов);
- простое и хорошо эксплуатируемое обслуживание символов (например, вызов символов кодами алфавитноцифрового знака, наложение символов друг на друга и т.д.);
- возможность встроенных алгоритмов для решения специальных задач (например, построение прямой, определенной двумя конечными точками);

- с точки зрения матобеспечения ЭВМ, высокий уровень обслуживания элементов и операций (например, хранение и перенос изображений в легко применяемой форме списков);
- простая заменяемость процедуры обмена данными между ЭВМ и терминалом;
- разнообразное изображение данных измерения в зависимости от времени и друг от друга (диаграмма, горизонтальная или вертикальная гистограмма, и т.д.).

Соответствующий вышеперечисленным требованиям терминал может быть создан встроением значительной интеллектуальности в устройство. Это осуществилось управлением с микропроцессором, программа которого запомнена в памяти ПППЗУ РЕПРОМ (такая программа называется "firmware").

В дальнейшем излагаются принципы оформления, осуществления трех составных частей hardware, firmware и software дисплейного терминала, потом приводятся несколько прикладных примеров.

2. УЗЛЫ СИСТЕМЫ ТЕРМИНАЛА ЭВМ

В первой фазе проектирования дисплейного терминала с микропроцессорным управлением нужно было тщательно рассмотреть вопрос разделения задачи между аппаратной (hardware), аппаратно-программной (firmware) и программной (software) системами.

Целесообразно разделить эту задачу на два вопроса:

- Что является оптимальным разделением задачи между ЭВМ и дисплейным терминалом?
- Из задач терминала что нужно решить аппаратно-программной системой и что - аппаратной?

Эти два вопроса изображены на рис. 3. В общем случае определение предела между отдельными "полями" не возможно; имеются однако некоторые факторы, на основе которых, в специальном случае, можно положить между.

2.1. Разделение задачи между ЭВМ и терминалом

Очевидно, что управление процессами не является задачей терминала, т.е. обмен данными между матобеспечением и терминалом представляет собой передачу информации, готовой для отображения на дисплейном терминале.

Важно, чтобы при составлении конкретных прикладных программ программист не должен был заниматься проблемой на многих уровнях. Другими словами, для пользователя *firmware* терминала должен быть полностью транспарентным. С точки зрения ЭВМ, терминал является оборудованием с особенно высокой интеллектуальностью.

На рис. 4 приведен круг задачи терминала и ЭВМ. На рис. показаны типичное сообщение о списках изображений, поступившее от ЭВМ и типичное сообщение о командах. Большим преимуществом является тот факт, что сложные задачи, связанные с изображением, могут быть формулированы в одной команде. Задачей ЭВМ являются в ходе составления, редактирования изображения сортировка, сжатие отдельных элементов (точек, символов, знаков, прямых) изображения для списка изображения. Полученные вышеуказанным образом списки изображений следует запоминать во внешней памяти, откуда, по желанию, в любое время можно их вызвать, воспроизводить, модифицировать. По сравнению с промышленными процессами построения изображений осуществляется в режиме офф-лайн, т.к. это - длительная операция, и относится к подготовительным работам. В режиме он-лайн (т.е. в ходе управления процессами) ЭВМ обновляет/дополняет постоянное изображение самыми новыми алфавитно-цифровыми данными измерения, изменениями гистограммы и символами, показывающими состояние. В таком слу-

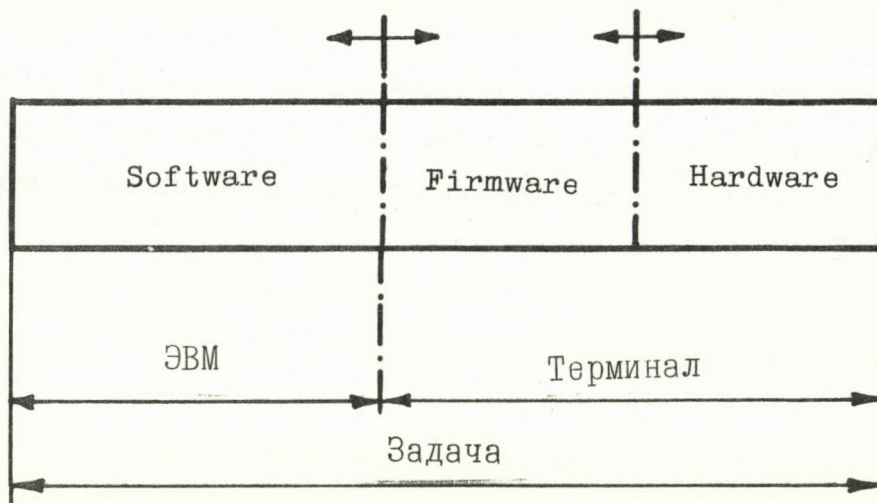


Рис. 3 Разделение Software/Firmware/Hardware

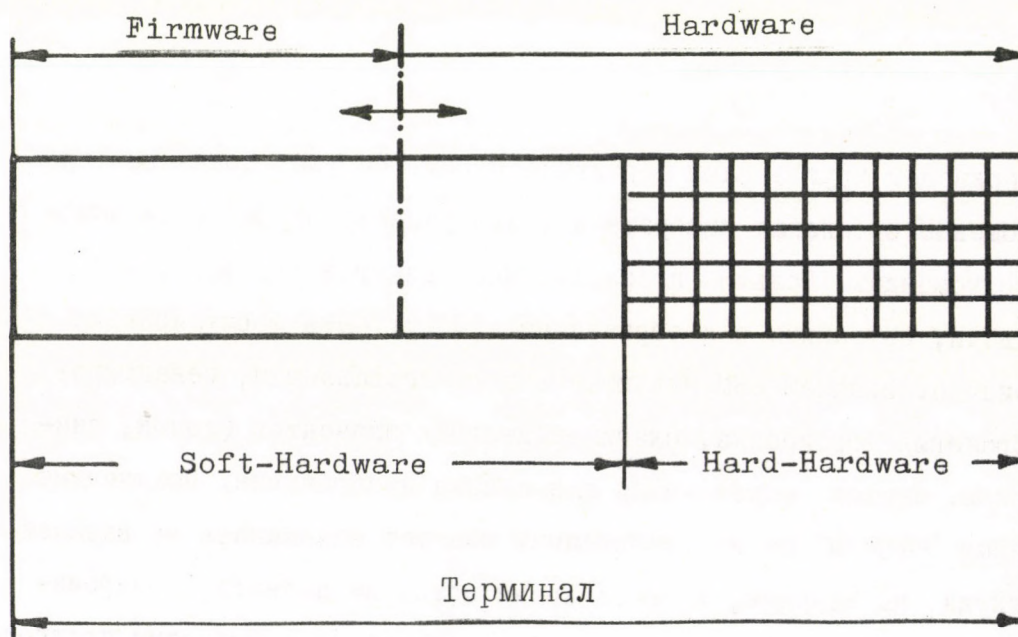


Рис. 4 Определение "Soft" и "Hard-Hardware"

чае терминал должен получить только изменяющуюся информацию изображения.

2.2 Разделение задачи между двумя узлами терминала

В терминале имеются такие функции, которые в никаких условиях не могут быть обслуживаны аппаратно-программной системой (например, такие как: сканирование, генерирование синхронных и видеосигналов ЭЛТ и т.д.). Эта часть терминала называется `hard-hardware`, а другие узлы, выполняющие все остальные функции, - которые могут быть аппаратной системой (`hardware`) или `firmware`, называется `soft-hardware` (см. рис.4).

Когда внутри `soft-hardware` хотим положить между `firmware-hardware` необходимо учитывать следующие факторы:

- ограничения в скорости и во внешнем тайминге, так как эти требования исключают возможность решения определенных функций с помощью `firmware`;
- стоимость аппаратной системы специальной цели нужно сравнивать со стоимостью памяти, необходимой для альтернативного решения с `firmware`;
- при производстве или при эксплуатации пользователем может возникать требование к изменению спецификаций, рабочих параметров без модификации аппаратной системы. Подходящий выбор предела между `firmware-hardware` дает возможность для создания различных терминальных семейств, или в большой мере может облегчить поиск ошибок и уход. Кроме этого, в соответствии со своими требованиями, потребитель получает возможность для изменения спецификаций.

2.3 Работа терминала

Основная конфигурация (без расширения) приведена на рис. 5. Оператор устанавливает маркерную точку на ту точку, где нужно построить или стирать элемент изображения. Потом нажимает на одну из клавиш функций, с помощью которой определяется операция. В случае надобности, нажимает кроме этого и на одну из алфавитно-цифровых клавиш. Каждое нажатие на клавишу вызывает прерывание в работе процессора и, таким образом, составляется, посылается к ЭВМ подходящее сообщение. В этом случае на экране терминала еще ничего не изменяется. При поступлении (приеме) сообщения - ЭВМ отражает его (в большинстве случаев), в списке элементов изображения осуществляется необходимая запись или необходимое стирание.

Отраженное сообщение интерпретируется в firmware терминала и выполняется отображение или стирание соответствующего элемента изображения. При окончании операции посылается ЭВМ подтверждение. Таким образом, оператор может составить изображение, состоящее из точек, линий, предварительно определенных символов и алфавитно-цифровых символов (последние могут быть "инверсными" т.е. на темной базе светлыми). Одновременно в ЭВМ образуется и составляется соответствующий список изображения, а именно, в "сжатой" форме. Оператор может наносить этот список изображения - как один файл - на магнитный диск или может вызвать нанесенный таким образом список изображения с диска, который ЭВМ посылает терминалу для отображения по блокам макс. 256 символов линии связи.

Для выполнения вышесказанной работы служат несколько специальных командных сообщений, такие например: гашение экрана дисплея, сброс и новый вызов списка изображения и т.д. Структура командных сообщений подобно структуре сообщений об элементах изображения с той разницей, что командные сообщения вообще не отражаются от ЭВМ.

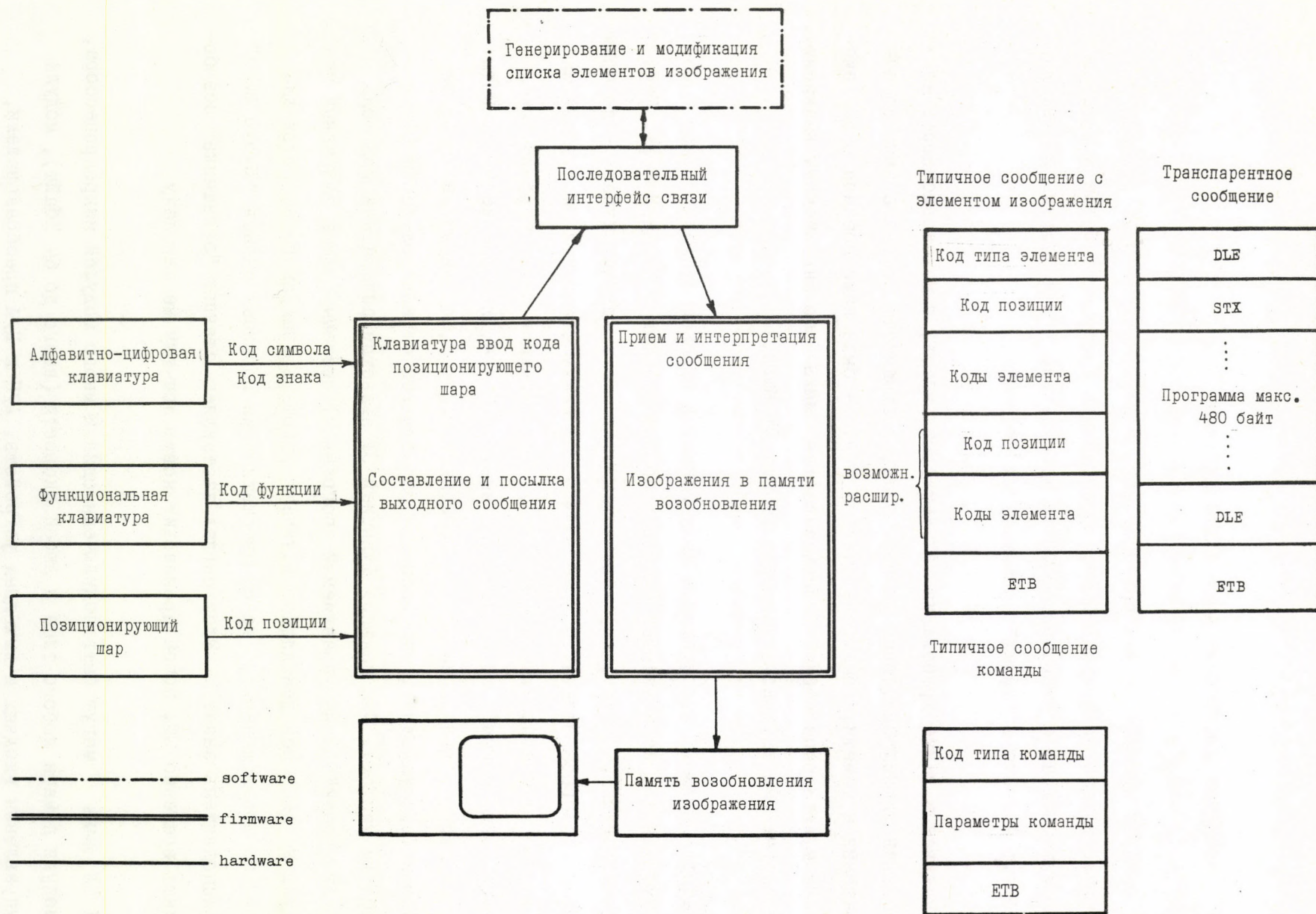


Рис. 5 Разделение задачи в системе ЭВМ и терминала

3. ПРИНЦИПЫ ПОСТРОЕНИЯ АППАРАТНОЙ СИСТЕМЫ

В области применения микропроцессоров, в первом этапе была создана соответствующая аппаратная база, позволяющая построение более и гораздо сложных систем с микропроцессорным управлением. Эти системы должны работать и в очень "умных" промышленных условиях.

Учитывая и такие требования как универсальность, т.е. возможность применения микропроцессоров разного типа, возможность простого прямого обращения к памяти, быстродействие и т.д. - была нами создана общая система мультипроцессорной шины (называется MMPS по английскому названию). В дальнейшем коротко излагается работа ОШ MMPS.

Работа ОШ MMPS основывается на перекрытой технике циклов поисков и обслуживания, увеличивающей скорость обмена данными в большой мере. Передача данных (или команды) осуществляется - в каждом случае - по принципу "вопроса/ответа" ("handshake") между станциями ОШ "задатчика" (master) и исполнителя (slave).). Теоретически любая станция может быть "задатчиком", но на практике имеются такие, которые не могут быть (например, станции памяти). Приоритет модулей, одновременно запрашивающих обслуживание, определяется расстоянием от контролера ОШ (физическим местом). Специальной функцией ОШ является аппаратный контроль "готовности" (и не управляемый программой) системы. Если выбранный модуль не способен немедленно начинать работать (например, не готов для приема новых данных), то он автоматически посылает сигнал "Slave Busy" (исполнитель занят). Этот сигнал заставляет станцию "задатчика" освободить временно ОШ; потом производит новую попытку на передачу.

К ОШ MMPS могут быть подключены один и много модулей микропроцессора, модули памяти любого типа и любой скорости (всего до 64 Кбайт), модули управления входных и выходных устройств, А/Ц и Ц/А преобразователи,

мультиплексоры, модули-интерфейсы ЭВМ, пульт управления оператора и т.д. Следует подчеркнуть, если в системе работает много микропроцессоров, то они должны друг с другом делить общее поле памяти.

Данная ОШ ММРС стала соответствующей базой аппаратной системы и для дисплейного терминала. Полная аппаратная система реализована на ОШ (см. рис. 6) и частично содержит модули общей цели, которые могут быть хорошо использованы и в других областях применения:

- модуль микропроцессор работает на ИС Intel 8008. Кроме узлов, необходимых для микропроцессора и ОШ содержит логику (схему) прерывания и два внешних регистра для облегчения программирования. (Эти регистры адресуются командами IOT ввода/вывода данных.) Полное количество адресируемых средств ввода-вывода - 128.
- Модуль ПЗУ/ЗУПВ содержит любую комбинацию элементов ПППЗУ, статической ЗУПВ и генератора символов. Полная адресная емкость одного модуля - 4 Кбайт.
- Модуль последовательного интерфейса связи, служащий для подключения терминала к ЭВМ через дуплексный асинхронный канал, работает по схеме UART со скоростью переноса до 9600 бод, устанавливаемой переключкой.
- Модуль клавишного интерфейса (интерфейса клавиатуры) работает с алфавитноцифровыми и функциональными клавишами в соответствии с этим посылает для процессора два различных вектора прерывания.
- Память для возобновления изображения представляет собой динамическую память с обращением с двух сторон и емкостью 8 Кбайт. Цикл восстановления содержания памяти запускается или внутренним тактовым генератором или внешним источником. При применении растрового телевизионного дисплея для возобнов-

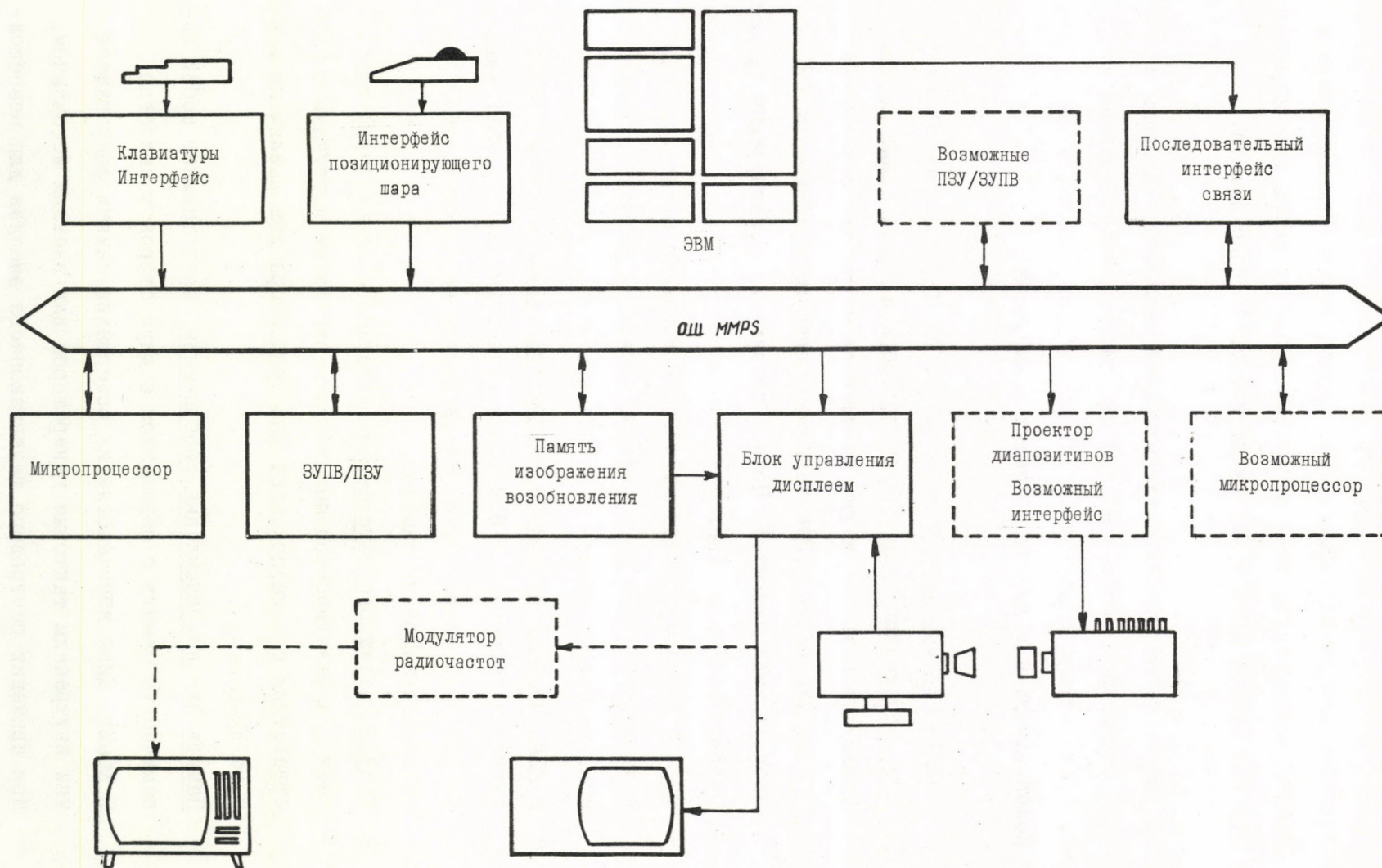


Рис.6 Функциональная блочная схема терминала

ления памяти служит горизонтальный синхронный сигнал. Для возобновления изображения, чтобы разгрузить ОШ, имеется возможность доступа к памяти и со стороны управления дисплеем. Система терминала дополнена двумя станциями ОШ специального назначения.

- Станция управления дисплеем по разрешению процессора считывает содержимое памяти возобновления (50 раз в сек) и преобразует его в телевизионный видео-сигнал, появляющийся на видеомониторе. В соответствии с емкостью памяти 8К байт (=64 Кбит) на экране имеются 256х256 адресируемых точек. Видео-вход модуля позволяет смешивать информацию внешнего изображения (камеру, видео-ленту) с изображением, получаемым из памяти возобновления.
- Позиционирующий шар служит для смещения точки маркера на экране. Оператор может легко устанавливать и переносить маркер на любую адресируемую точку экрана. Программа микропроцессора способна считывать содержимое регистров позиции или записывать в них. (Рис.6).

Благодаря организации ОШ и модульному построению, к системе легко можно подключить устройства расширения. В случае применения, когда отображаемый рисунок состоит из постоянной и изменяющейся частей, является преимущественным, если не запоминать полное изображение в памяти. Вместо этого, постоянное содержимое изображения можно запомнить на диапозитивах и проектировать в телевизионную камеру, выход которой устройство управления смешивает с изображением из памяти воспроизведения. Таким образом освобождается значительная часть емкости внешней памяти и нагрузка связи между ЭВМ и терминалом уменьшается в значительной мере. Для решения вышеуказанного способа необходимо иметь проектор карусельного типа, управляемый простым модулем интерфейса, для адресации диапозитива.

В том случае, если возникает требование к особенно большому быстродействию, можно применять и второй модуль микропроцессора. В двухпроцессорной системе задачи могут быть разделены, например, так: один из процессоров управляет вводом и выводом информации (через канал связи от клавиатуры и т.д.), а другой генерирует изображение и модифицирует его. Конечно, микропроцессоры работают одновременно и параллельно. Так как оба микропроцессора работают на общем поле памяти, не имеется надобности в прямом обмене данными между двумя процессорами.

4. ПОСТРОЕНИЕ И РАБОТА FIRMWARE

Схема firmware приведена на рис. 7. Так как эта блок-схема довольно упрощена, несколько дополнительных объяснений могут оказать помощь в понимании работы терминала.

Имеются 5 входных точек прерывания, эти точки видны на рисунке над пунктиром. В случае прерывания, новые прерывания автоматически запрещаются, выполняется необходимая операция и прерывание снова разрешается. После такой операции выполняется возврат к исходной программе, которая по сущности является бесконечной петлей ожидания.

Входные точки прерывания следующие:

ВКЛЮЧЕНИЕ-СТАРТ - стирается память возобновления, устанавливаются определенные регистры и триггеры на исходное значение; прерывание разрешается и программа переходит в петлю ожидания.

ФУНКЦИОНАЛЬНАЯ КЛАВИАТУРА - прежде всего определяется, которая из функциональных клавиш нажата. В том случае, если нажатая клавиша не служит для генерирования символа или знака, то полное сообщение составляется в выходной памяти и поступает в ЭВМ.

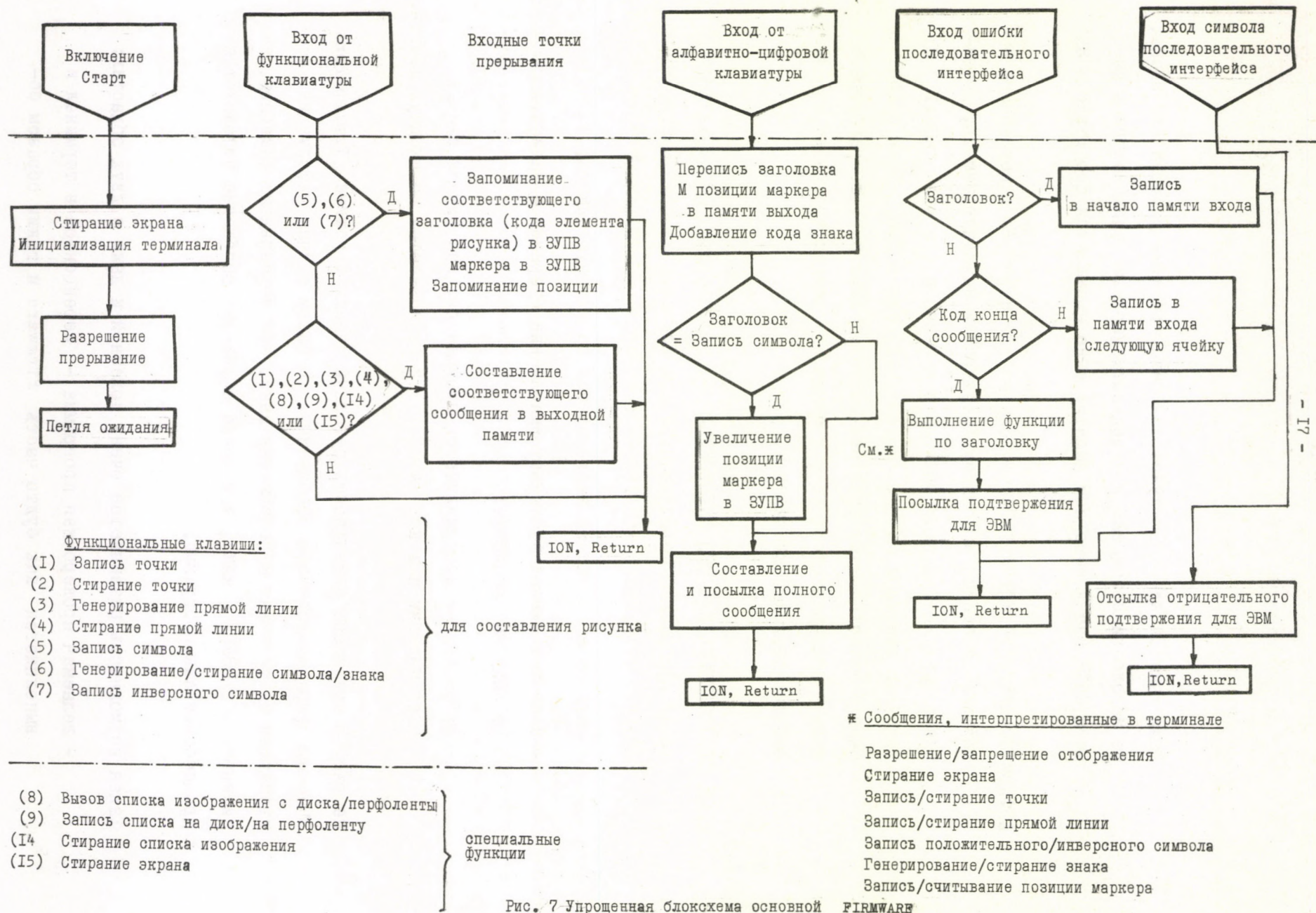


Рис. 7 Упрощенная блок-схема основной FIRMWARE

Если нажатая клавиша служит для генерирования символа или знака, то заголовки сообщений и координаты маркера запоминаются в памяти ОЗПВ (не в выходной памяти) и не посылается сообщение в ЭВМ.

АЛФАВИТНО-ЦИФРОВАЯ КЛАВИАТУРА - служит для переноса запомненного заголовка сообщения и позиции "маркера" в выходную память, дополняя алфавитно-цифровым символическим кодом (в коде ASCII) и конечным символом (ЕТВ), потом поступает в ЭВМ. Если сообщение было о представлении буквенно-цифровых символов, то регистры маркера увеличиваются на одну позицию для следующего ввода от алфавитно-цифровой клавиатуры.

ВВОД ОТ ПОСЛЕДОВАТЕЛЬНОГО ИНТЕРФЕЙСА - представляет собой входную точку информации (отображенной или исходной из ЭВМ).Прежде всего следует определить тип входного кода:

- заголовок сообщения записывается во входную память;
- конечный символ запускает выполнение операции в зависимости от символа заголовка;
- любое другое воспринимается программой, как символ данных (код позиции или код ASCII) и записывается во входную память.

На рис. 5 приведена возможность передачи данных в режиме Transparent Message (транспарантного сообщения). В таком случае информация, поступающая на терминал и содержащая макс. 480 8-разрядных символов, записывается в ОЗУПВ (RAM), и в конце сообщения сообщение выполняется процессором как программа.

Такая возможность оказывается очень полезной и даже в двух областях:

- желаемая проверочная программа - переносится в терминал и выполняется как будто часть firmware и таким образом об-

легчается поиск ошибок и проверка терминала (полная длина проверочной программы может быть больше, чем длина встроенной firmware).

- функции не интерпретированные firmware (что для hardware вполне возможно) могут быть декларированы переносом соответствующей firmware из ЭВМ, например: разрешение или запрещение смещения маркера или мигание точек кадра и т.д.

ВХОД ОШИБКИ ОТ ПОСЛЕДОВАТЕЛЬНОГО ИНТЕРФЕЙСА - в случае ошибки в формате или в четности программа засылает в ЭВМ отрицательное подтверждение работы.

Размер основного варианта вышеприведенной firmware прибл. 3,5 Кбайт.

5. ПОСТРОЕНИЕ И РАБОТА ПРОГРАММНОЙ СИСТЕМЫ

Подключение терминала к ЭВМ осуществляется через последовательный асинхронный канал, так ЭВМ может быть любого типа. В ЦИФИ терминал подключен к малой ЭВМ ТРА-I и операционная программа написана и на языке ТРА-I. Программа - так называемая программа редактирования изображения терминала. Ее задачей является дать работу терминалу по сравнению с управлением процессами в режиме офф-лайн. В это время составляются рисунки, диаграммы, таблицы, необходимые для управления процессами.

Конфигурация, требуемая программой: малая ЭВМ ТРА-I с емкостью памяти 8К слов, перфоленточные периферийные устройства и внешний накопитель на магнитных дисках.

Элементы рисунка, построенного оператором, запоминаются программой в

списке элементов изображения. Блоки элементов изображения, поступающие из терминала - после их отображения, соответствующей трансформацией помещаются в соответствующей части списка. По команде, запрашивающей посылку полного списка, программа засылает терминалу отдельные части списка элементов изображения по блокам макс. с 256 символами. До подтверждения полной обработки блока ЭВМ не может послать новый блок. Из ОЗУ можно вынести на накопители полный список изображения и потом, в случае надобности, возвращать его. Возможные носители данных - перфолента или магнитный диск. Блок-схема программы приведена на рис. 8. Эта блок-схема подобно блок-схеме firmware показывает упрощенную блок-схему software. В дальнейшем коротко излагается работа программы.

После запуска программа выполняет начальные функции установки, инициализирует терминал, потом поступает в основной цикл, который представляет собой петлю ожидания и здесь только контролирует, поступил-ли полный блок (имеется ли на конце ЕТВ). В случае прерывания, вызванного входным или выходным символом линии, выходит из петли.

Блоки, поступившие из терминала, содержат элемент изображения или команду. После их обработки программа переходит снова в исходный цикл.

Блоки элементов изображения могут вызвать запись или стирание. При стирании элемента определенной позиции (точка, прямая, символ) происходит сжатие списков. Стирание буквенно-цифрового символа или инверсного буквенно-цифрового символа осуществляется кодом SPACE (промежутка между символами) так, что на место стертого символа записывается SPACE, значит не происходит сжатие списка.

Различные элементы изображения записываются в различной форме. Точка определяется двумя координатами, прямая - четырьмя координатами, двух конечных точек в списке.

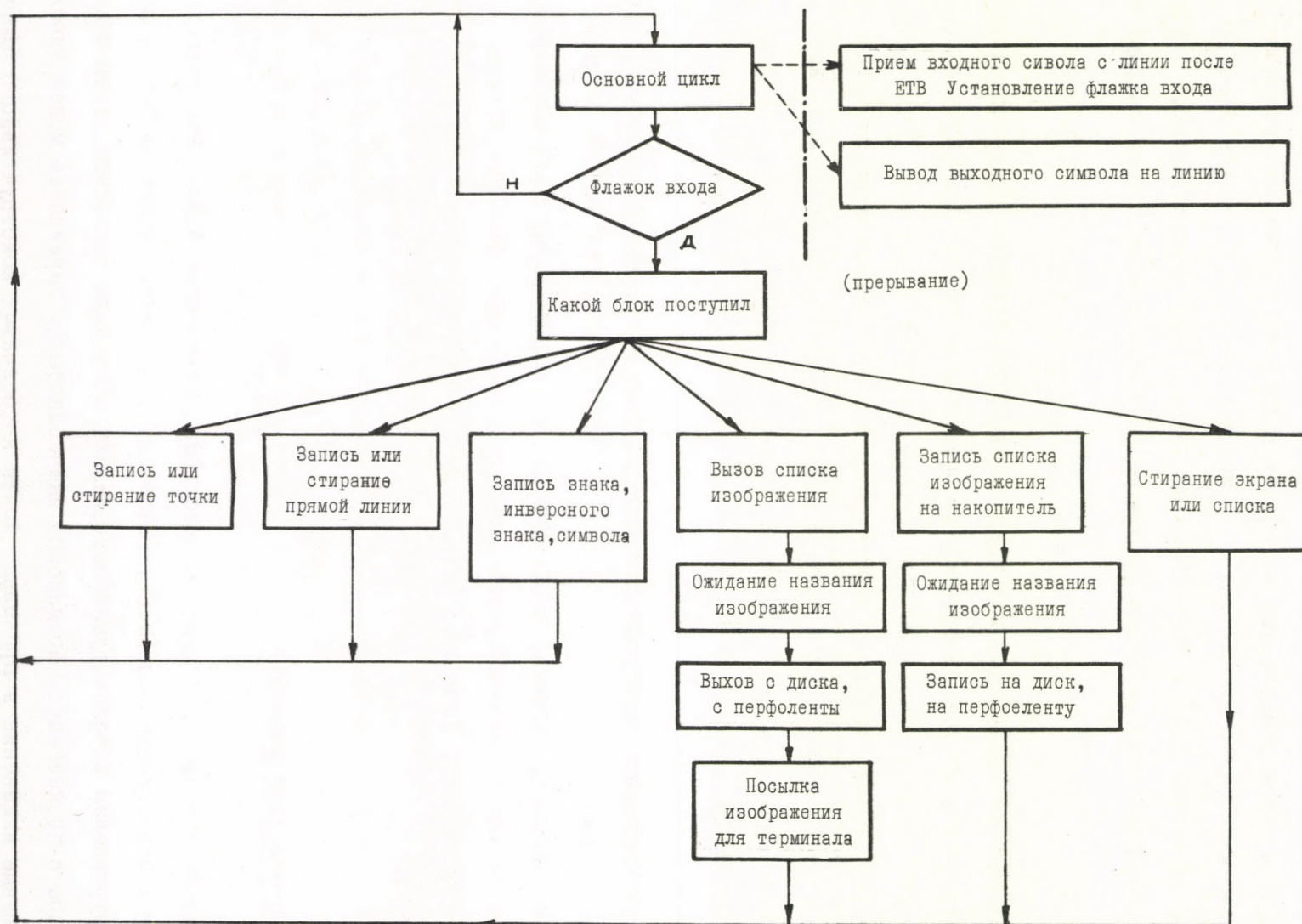


Рис. 8 Блоксхема матобеспечения ЭВМ

Буквенно-цифровой символ и инверсный буквенно-цифровой символ определяются координатами только в том случае, если в позиции не следуют друг за другом. Если они следуют по позициям друг за другом, для их определения необходимо всего один код символа вместо трех байтов (две координаты и один код) и таким образом список будет более сжатым, - разумеется, перед первым кодом находится одна единственная пара координат начальной позиции.

При записи на накопители или вызове списка изображения нажатием соответствующей функциональной клавиши оператор переносит программу в состояние "ожидания названия" (на экране появляется надпись: FILE NAME). Оператор задает название, состоящее макс. из 4 знаков и на их конце код "каретка обратно" или код CTRL-P. "Каретка обратно" запускает операцию записи на диск или вызов отсюда, а другой код - операцию с перфолентой.

Список полученный программой редактирования изображения, преобразуется подпрограммой DIO (Disc Input-Output-ввод-вывод на магнитный диск) в такой формат, который необходим для программы управления процессами. В случае различных операционных систем следует заменять только эту подпрограмму (DIO), чтобы получить соответствующий список изображения.

6. ПРИКЛАДНЫЕ ПРИМЕРЫ

Дисплейный терминал с микропроцессорным управлением эффективно применяется в большинстве областей управления процессами. Гибкие свойства, высокоуровневая интеллектуальность делают идеальным средством связи оператора с процессами. Применение вышеизложенных терминалов может вносить основные изменения в принципы почти необозримых конструктивного оформ-

ления пультов управления сложных процессов. Большинство аналоговых и цифровых индикационных приборов, сигнальных ламп становятся, таким образом, излишними; отображение функциональных схем полного процесса или отдельных частей процесса облегчает контроль и слежение за событиями. Информация, изображаемая на хорошо отредактированных рисунках на экране не окажется переполненной, даже при многочисленных измерительных результатах. Информация измерения гораздо большего количества в легко обозримой форме можно довести до сведения оператора с возможностью вызова друг за другом схем процесса.

В качестве примера хотим представить три способа применения из различных областей. На рис.9 приведен пример из области управления реактором. Контролируемое устройство - схема первичного и вторичного контура охлаждения реактора - хорошо обозримое; гистограмма на левой нижней стороне рисунка наглядно показывает изменение состояния измеряемой температуры. Инверсные алфавитно-цифровые символы или мигание элементов изображения обращают внимание оператора на переход критических предельных величин. Измеренные значения появляются и вне гистограммы на функциональной схеме при точке действительного измерения.

На рис. 10 приведен пример из области нефтепромышленности - узел сети трубопроводов. Хорошо видно, например, простое символическое обозначение закрытого, полузакрытого или открытого состояния исполнительных органов (вентилей или затворов).

На рис. 11 показана подстанция сети разделения электрической энергии. Здесь при данных напряжения и мощности видно действительное состояние прерывателей.

До сих пор применяются два таких терминала. Система сбора данных измерения Притиссайской Теплоэлектростанции применяется при содействии от-

дела техники управления электростанцией Исследовательского Института Электрической Энергии (VEIKI).

Система управления трубопроводов нефтяных продуктов при помощи ЭВМ у предприятия АФОР была разработана при содействии основного предпринимателя Комбината Автоматики ММГ.

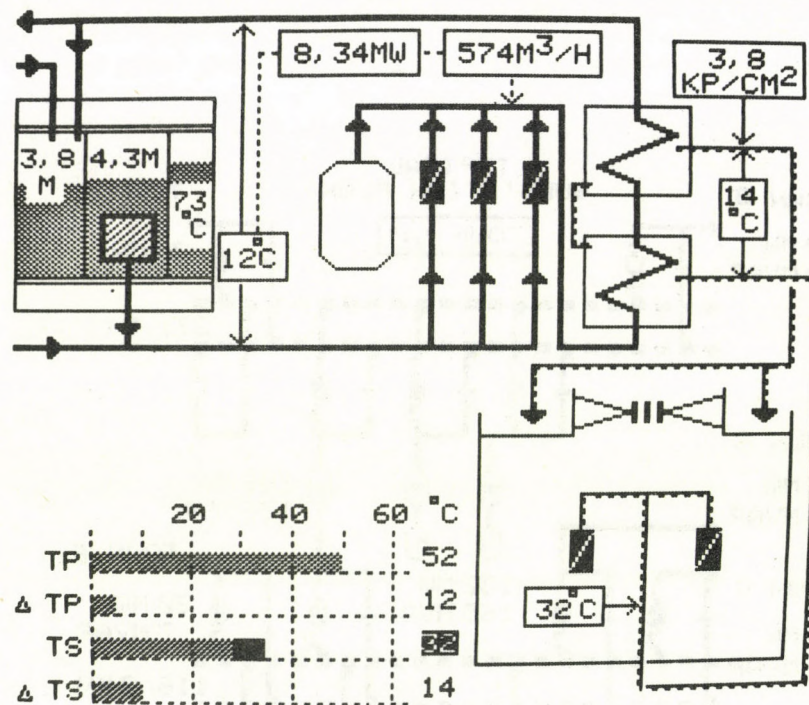


Рис. 9

CI 1.56 *SL *ZA *ES 1643 57.67

08SSZ * *

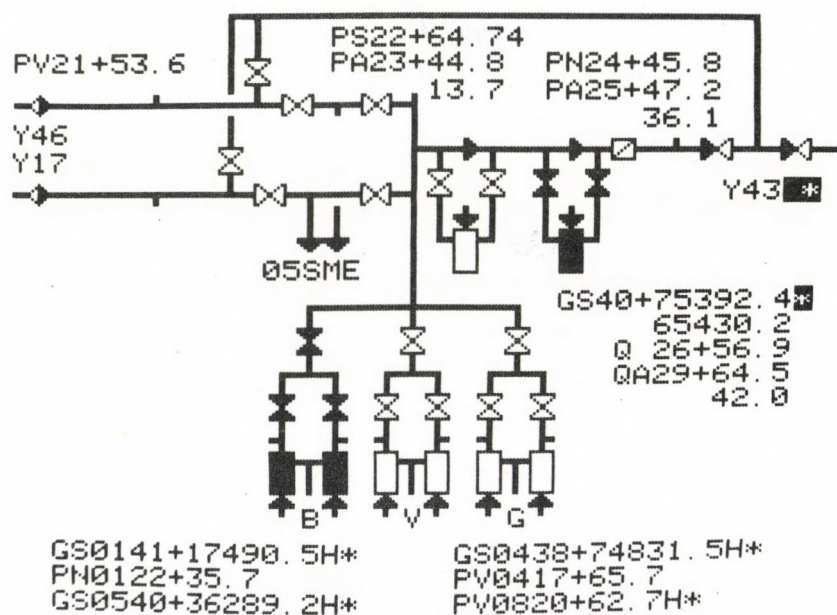


Рис. 10

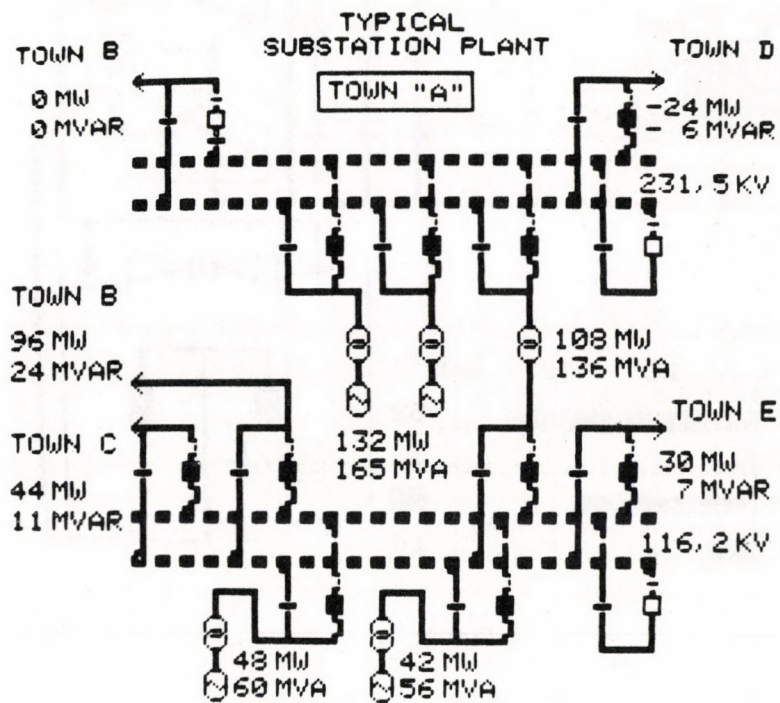


Рис. 11

Л и т е р а т у р а

1. Gy. Ambrózy, I. Rényi, F. Vajda:
Design and Application of an LSI Multi-Microprocessor Based Bus
and Module System
Jurema 1976. April 1976, Выпуск конференции
2. J. Miskolczi, T.L. Sándor, F.Törő:
An LSI Microprocessor Controlled TV Raster Display Terminal for
Process Control Interactive Flowcharting
Jurema 1976. April 1976, Выпуск конференции
3. I. Rényi, F. Vajda:
A Microprocessor Controlled Display Terminal for Interactive
Flowchart Generation
EUROMICRO 1976. Velence, Выпуск конференции
4. Д. Папп: Применение дисплея с управлением микропроцессором типа
КФКИ PE-I2 в технике управления ТЭС
RTF No.5/76. Сообщение об исследованиях VEIKI, март 1976 г.
5. Д. Папп, Л. Шандор, Ф. Тэрэ:
Применение дисплейного терминала с управлением микропроцессором
при задачах управления в области электропромышленности
XIII.Симпозиум электрического измерения и регулирования в области
промышленности электрической энергии,
Балатонсеплак, сентябрь 1976.

62.409



Kiadja a Központi Fizikai Kutató Intézet
Felelős kiadó: Sándory Mihály igazgató
Szakmai lektor: Láng István
Nyelvi lektor: Erényi István
Példányszám: 360 Törzsszám: 77-422
Készült a KFKI sokszorosító üzemében
Budapest, 1977. április hó